

(19)

Europäisches Patentamt

European Patent Office

Office européen des brevets



(11)

EP 0 810 526 A1

(12)

## EUROPEAN PATENT APPLICATION

(43) Date of publication:  
03.12.1997 Bulletin 1997/49

(51) Int Cl. 6: G06F 11/14

(21) Application number: 97303680.9

(22) Date of filing: 02.06.1997

(84) Designated Contracting States:  
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC  
NL PT SE

- Satyanarayanan, Ottalingam  
Fremont, California 94539 (US)
- Pruscino, Angelo  
Mountain View, California 94040 (US)

(30) Priority: 31.05.1996 US 656386

(74) Representative:  
Cross, Rupert Edward Blount et al  
BOULT WADE TENNANT,  
27 Furnival Street  
London EC4A 1PQ (GB)(71) Applicant: SUN MICROSYSTEMS, INC.  
Mountain View, CA 94043 (US)(72) Inventors:  
• Molin, Hossein  
San Francisco, California 94131 (US)

## (54) Quorum mechanism in a two-node distributed computer system

(57) Each node of a failing distributed computer system, e.g., as a result of a split-brain failure, races to achieve a quorum by successfully reserving two shared storage devices which are designated quorum controllers. During normal operation of the distributed computer system, each of the quorum controllers is associated with and reserved by a respective node. During the race for quorum in response to a detected failure of the distributed computer system, each node which has not failed forcibly reserves the quorum controller which is associated with the other node. If node simultaneously holds reservations for both quorum controllers, that node has acquired a quorum. The forcible reservation of a shared storage device does not fail even if another node holds a valid reservation to the same storage device. Accordingly, a failed node which does not relinquish a reservation to the node's quorum controller cannot prevent another node from acquiring a quorum. Prior to forcibly reserving the quorum controller of another node, each node verifies that it continues to hold a reservation of the node's own associated quorum controller. If a node no longer holds a reservation of the node's own associated quorum controller, that node has lost the race for quorum since another node has already forcibly reserved the former node's associated quorum controller. Thus, quorum can be efficiently and effectively determined by independent nodes of a failing distributed computer system notwithstanding the failure of a failing node to relinquish shared storage device reservations held by the failing node.

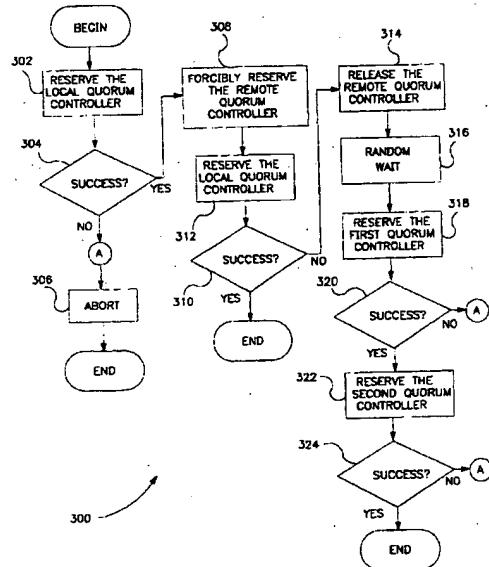


FIG. 3

EP 0 810 526 A1

**Description****FIELD OF THE INVENTION**

The present invention relates to fault tolerance in distributed computer systems and, in particular, to a particularly robust mechanism for determining which of two nodes in a failing distributed computer system has exclusive access to shared resources.

**BACKGROUND OF THE INVENTION**

The complexity of tasks performed by computers today continues to grow rapidly. Accordingly, the importance of distributed computer systems, in which computer applications and databases are distributed over multiple computers, is increasing dramatically.

In a distributed computer system, multiple constituent computers of the distributed computer system, which are generally referred to as "nodes" or "hosts," have shared access to storage devices which collectively store a distributed database. Failure of a component of the distributed computer system can cause corruption of the distributed database and, therefore, potential loss of valuable and perhaps irreplaceable information stored in the distributed database. The following example is illustrative.

Suppose two nodes of a distributed computer system continue to operate normally, but a communications link between the two, through which the nodes coordinate access to the shared storage devices, fails. Each of the two nodes may continue to access the distributed database although shared access to the database cannot be coordinated. If both nodes continue to access the distributed database, the distributed database can very likely become corrupted and loss of valuable information can easily result. This is commonly known as the "split-brain" problem.

The following example illustrates the split-brain problem in more detail. Suppose that two separate computers are used to maintain bank records in a distributed database. Suppose further that two people, both of whom have access to a single account, deposit money into the account at substantially the same time, wherein each deposit transaction is recorded in the distributed database by a respective one of the two computers. Suppose further still that all communications between the two computers have failed. Each of the computers records the deposit by retrieving the current balance of the account, e.g., \$200, and storing as the new balance the sum of the previous balance and the deposit amount, e.g., \$500 by one person and \$100 by the other person. If the computers access the distributed database at different times, each deposit is likely to be accurately recorded in the distributed database. However, if each computer retrieves the current balance at approximately the same time, each determines the current balance to be \$200 and replaces the recorded balance

with either \$700 or \$300. The new balance is then either \$700 or \$300 depending on which of the computers is the last to record the new balance, and the loss of the deposit remains undetected. However, if only one of the computers is allowed access to the distributed database, failure to record either deposit is noted and remedial action can be taken.

In this illustrative example, substantive information regarding a single transaction is lost. It is possible in some circumstances to lose information regarding inter-relationships regarding information stored in the distributed database, e.g., information regarding the location of various records of the distributed database. If such information is lost, significant portions of the distributed database can become irretrievable. The split-brain problem is therefore a serious problem with distributed databases and should be avoided.

There are generally two classes of solutions to the split-brain problem: solutions involving two-node distributed computer systems and solutions involving distributed computer systems having more than two nodes. In the latter class, a failed communications link can result in two groups of nodes which cannot communicate with one another. One conventional solution is to grant access to shared resources, e.g., shared storage devices, to the group containing a simple majority of the nodes of the distributed computer system. The group having such access to the shared storage devices is generally referred to as having attained a quorum. Each group can determine the number of nodes in the group and only one group can contain a simple majority of all nodes of the distributed computer system. The group containing less than a majority voluntarily refrains from accessing shared resources. If each group includes exactly one-half of the nodes of the distributed computer system, the problem becomes analogous to a split-brain problem involving a two-node distributed computer system and such a solution is generally employed.

In a two-node distributed computer system, the split-brain problem can be resolved by involving the two nodes in a race to reserve as many shared resources as possible. Such a race is generally referred to as a race for quorum. Each node attempts to reserve all shared storage devices. If a node successfully reserves a simple majority of all shared storage devices, that node has attained a quorum and has access to all shared devices. Conversely, if a node fails to reserve a simple majority of all shared storage devices, that node voluntarily refrains from accessing any shared storage device. Thus, if a node of the two-node distributed computer system fails, the remaining node successfully reserves a majority of shared storage devices and continues to manage the distributed database by access to the shared storage devices. In addition, if all communications links between the two nodes of a two-node distributed computer system fail, the two nodes race to reserve the shared storage devices, and the node which wins the race for quorum continues to manage the dis-

tributed database by accessing the shared storage devices while the node which loses the race for quorum voluntarily refrains from accessing any shared storage device to avoid corruption of the distributed database.

Use of shared storage device reservations as a race parameter becomes infeasible when reservation of such shared storage devices is a necessary part of access to such shared storage devices. For example, if a node of a two-node distributed computer system fails, reservations held by the failing node may not be relinquished. As a result, the remaining node, which has not failed, may not be able to reserve a majority of shared storage devices and, accordingly, may not continue to operate normally, i.e., may voluntarily refrain from accessing the shared storage devices in the mistaken belief that the remaining node has lost the race for quorum. Such is even more likely in conventional distributed computer systems which designate a single shared storage device as a quorum controller, reservation of which constitutes winning the race for quorum. A failing node may hold, and fail to relinquish, a reservation of the quorum controller. When the non-failing node cannot attain quorum, the distributed computer system fails, and such failure is unnecessary since the remaining node is otherwise generally capable of continuing to operate normally. A failed node can retain device reservations when the failure is software related, e.g., either the operating system of the node has failed or the computer process accessing or managing the distributed database has failed. When the operating system has failed, human interaction is generally required to bring the node to a state in which the held device reservations are relinquished. When a computer process other than the operating system has failed, the failure may go undetected by the operating system.

What therefore remains as an unsolved need in the industry is a quorum mechanism which does not require that a failed node relinquish reservations of shared storage devices.

#### SUMMARY OF THE INVENTION

In accordance with the present invention, each node of a failing distributed computer system races to achieve a quorum by successfully reserving two shared storage devices which are designated quorum controllers. During normal operation of the distributed computer system, each of the quorum controllers is associated with and reserved by a respective node. During the race for quorum in response to a detected failure of the distributed computer system, each node which has not failed forcibly reserves the quorum controller which is associated with the other node. If a node simultaneously holds reservations for both quorum controllers, that node has acquired a quorum. The forcible reservation of a shared storage device does not fail even if another node holds a valid reservation to the same storage device. Accordingly, a failed node which does not relin-

quish a reservation to the node's quorum controller cannot prevent another node from acquiring a quorum.

Prior to forcibly reserving the quorum controller of another node, each node verifies that it continues to hold a reservation of the node's own associated quorum controller. If a node no longer holds a reservation of the node's own associated quorum controller, that node has lost the race for quorum since another node has already forcibly reserved the former node's associated quorum controller. Thus, in accordance with the present invention, quorum can be efficiently and effectively determined by independent nodes of a failing distributed computer system notwithstanding the failure of a failing node to relinquish shared storage device reservations held by the failing node.

If a node has forcibly reserved the quorum controller of another node but no longer holds a valid reservation to the node's own quorum controller, the race for quorum has resulted in a tie. Both nodes of the distributed computer system must be operational for a tie to result. Therefore, there is no chance that a node can fail to relinquish a reservation of a shared storage device. Accordingly, a second race for quorum is used to resolve the tie between the nodes of the distributed computer system. Prior to the second race for quorum, each node relinquishes all reservations to any quorum controller such that no quorum controller is reserved by any node. In the second race for quorum, each node attempts to reserve a selected one of the quorum controllers in a predetermined sequence. The node which successfully reserves the selected quorum controller wins the second race for quorum and can access any of the shared storage devices. The other node determines that it has not acquired a quorum and voluntarily refrains from accessing any shared storage device.

Since force reservations are used in the first race for quorum, a node's failure to relinquish a reservation to a quorum controller cannot prevent another node from acquiring a quorum and, accordingly, subsequent access to the shared storage devices of the distributed computer system. The present invention therefore represents a significantly more robust and fault-tolerant mechanisms for determining quorum in a failing distributed computer system than prior art mechanisms. In addition, since each node has a quorum controller reserved during normal operation, loss of such a reservation during a race for quorum indicates that another node has forcibly reserved the quorum controller and is therefore operative. As a result, a second race for quorum which requires that each node relinquish reservations of quorum controllers can be used to resolve quorum when the first race for quorum results in a tie.

#### BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of a two-node distributed computer system in accordance with the present invention.

Figure 2 is a block diagram of the two-node distributed computer system of Figure 1 and shows computer processes executing within each of the nodes of the distributed computer system.

Figure 3 is a logic flow diagram of the determination by each node regarding which of the nodes acquires quorum in a fault condition.

Figure 4 is a logic flow diagram of the processing of a node of the distributed computer system of Figures 1 and 2 upon determination that quorum is not acquired.

#### DETAILED DESCRIPTION

Generally, two shared storage devices, e.g., shared storage devices 112A and 112B (Figure 2), of a two-node distributed computer system 100 are designated quorum controllers and are used in a novel quorum acquisition process, which is described more completely below. Briefly, each of nodes 100A-B of distributed computer system 100 detect a failure of the other node and performs the quorum acquisition process to form consensus regarding which of nodes 100A-B has acquired a quorum and which therefore has access to shared storage devices 112A-D and can therefore continue to operate. Each node has a respective one of the quorum controllers reserved during normal operation. A node acquires a quorum by reserving both quorum controllers. A node reserves a quorum controller which is normally reserved to another node by using a forcible reservation, which is described more completely below. The forcible reservation of a shared storage device does not fail even if another node holds a valid reservation to the same storage device and therefore prevents a failed node from preventing another node to acquire a quorum.

Appreciation of the present embodiment is facilitated by the following description of the hardware and software components of two-node distributed computer system 100 (Figure 1). Distributed computer system 100 includes a node 100A and a node 100B. Nodes 100A-B are directly analogous to one another and the following description of node 100A is equally applicable to node 100B.

Node 100A includes one or more processors 102A, each of which retrieves computer instructions from memory 104A through a bus 106A and executes the retrieved computer instructions. In executing retrieved computer instructions, each of processors 102A can retrieve data from and write data to memory 104 and any and all of shared storage devices 112A-D through bus 106. Memory 104 can include any type of computer memory including, without limitation, randomly accessible memory (RAM), read-only memory (ROM), and storage devices which use magnetic and/or optical storage media such as magnetic and/or optical disks. Shared storage devices 112A-D are each a storage device or an array of storage devices which can be simultaneously coupled to two or more computers. As shown in Figure

, 1, each of shared storage devices 112A-D is coupled both to bus 106A of node 100A and to bus 106B of node 100B. Each of shared storage devices 112A-D is accessed by each of nodes 100A and 100B as a single device although each of shared storage devices 112A-D can be an array of storage devices. For example, any of shared storage devices 112A-D can be a SPARC Storage Array available from Sun Microsystems, Inc. of Mountain View, California.

- 5
- 10
- 15
- 20
- 25
- 30
- 35
- 40
- 45
- 50

Each of shared storage devices 112A-D can be reserved by either node 100A or node 100B. For example, any of processors 102A can issue control signals through bus 106 to shared storage device 112C which cause reservation of storage device 112C. In response to the control signals, shared storage device 112C determines whether shared storage device 112C is already reserved as represented in the physical state of shared storage device 112C, e.g., in the state of a flag or an identification of the holder of the current reservation as represented in a register of shared storage device 112C. If shared storage device 112C is not currently reserved, shared storage device 112C changes its physical state to indicate that shared storage device 112C is now reserved by node 100A. Conversely, if shared storage device 112C is currently reserved, shared storage device 112C sends through bus 106 to processors 102A signals which indicate that the attempted reservation is refused.

In addition, each of processors 102A can issue control signals to a network access device 108A which cause network access device 108A to transfer data through network 110 between network access device 108A of node 100A and network access device 108B of node 100B in a conventional manner. In one embodiment, network 110 is the well-known Ethernet network and network access devices 108A and 108B are conventional Ethernet controller circuitry.

Distributed computer system 100 is represented in simplified form in Figure 2. Executing within node 100A are the following computer processes: (i) distributed database supervisor 202A, (ii) operating system 204A, and (iii) distributed application 206A. Similarly, distributed database supervisor 202B, operating system 204B, and distributed application 206B execute within node 100B and are directly analogous to distributed database supervisor 202A, operating system 204A, and distributed application 206A, respectively. Accordingly, the following description of distributed database supervisor 202A, operating system 204A, and distributed application 206A is equally applicable to distributed database supervisor 202B, operating system 204B, and distributed application 206B, respectively.

Distributed application 206A cooperates with distributed application 206B to manage a distributed database which is stored in one or more of shared storage devices 112A-D. Distributed application 206A manages the distributed database by storing data in and retrieving data from the distributed database and by maintaining

metadata within the database. Metadata generally refers to data within a database which represents a structure of substantive data within the database, frequently representing inter-relationships between substantive data stored in the database. In one embodiment, distributed application 206A is the Oracle Parallel Server available from Oracle Corporation of Redwood Shores, California.

Prior to accessing any data of the distributed database, e.g., a particular block of data of the distributed database, distributed application 206A requests from distributed database supervisor 202A exclusive access to the particular block of data. Distributed database supervisor 202A coordinates with distributed database supervisor 202B to acquire exclusive access to the particular block of data and indicates through reply signals to distributed application 206A whether distributed application 206A has exclusive access to the particular block of data. If distributed application 206A successfully acquires exclusive access to the particular block of data, distributed application 206A can store data in the particular block of data without simultaneous access by node 100B and therefore without risking corruption of the distributed database.

Distributed database supervisor 202A communicates and cooperates with distributed database supervisor 202B to acquire and relinquish exclusive access for distributed application 206A to specific blocks of data stored in shared storage devices 112A-D. Distributed database supervisors 202A-B communicate through communications link 210, which includes network access devices 108A-B (Figure 1) and network 110. If node 100B (Figure 2) or communications link 210 fails, distributed database supervisor 202A (Figure 2) detects the failure and reports the failure to operating system 204A. Failure detection by distributed database supervisor 202A is described more completely below.

In response to detection of a failure of communications link 210 or node 100B, distributed database supervisor 202A attempts to acquire a quorum according to the steps shown in logic flow diagram 300 (Figure 3). If distributed database supervisor 202A (Figure 2) successfully acquires a quorum according to logic flow diagram 300 (Figure 3), distributed database supervisor 202A (Figure 2) (i) assumes that distributed database supervisor 202B can no longer access any of shared storage devices 112A-D and (ii) allows distributed application 206A to continue to access shared storage devices 112A-D. Conversely, if distributed database supervisor 202A fails to acquire a quorum according to logic flow diagram 300 (Figure 3), distributed database supervisor 202A (Figure 2) (i) assumes that node 100B has not failed and has acquired a quorum and (ii) voluntarily refrains from granting to distributed application 206A access to shared storage devices 112A-D since exclusive access cannot be guaranteed.

Processing according to logic flow diagram 300 (Figure 3) presumes that a respective one of shared

storage devices 112A-D (Figure 2) has been designated as a quorum controller for each of nodes 100A and 100B. For example, shared storage device 112A can be designated as the quorum controller for node 100A and shared storage device 112B can be designated as the quorum controller for node 100B. Generally, any shared storage device can be designated as the quorum controller for a node of distributed computer system 100 (Figure 1) as long as the following criteria are met. First, 10 a physically different shared storage device must be designated as a quorum controller for each node. Second, the quorum controller for a particular node must be reserved to the node during normal operation of the node. Third, the quorum controller for each node must be 15 physically shared by the nodes of distributed computer system 100. In the illustrative example described herein, (i) shared storage devices 112A and 112B are physically different devices, (ii) shared storage devices 112A and 112B are reserved by nodes 100A and 100B, 20 respectively, during normal operation of nodes 100A and 100B, respectfully, and (iii) shared storage devices 112A and 112B are physically shared by nodes 100A and 100B. Accordingly, designation of shared storage devices 112A and 112B as quorum controllers for nodes 25 100A and 100B, respectively, is an appropriate designation.

Processing according to logic flow diagram 300 (Figure 3) further presumes that the quorum controllers for nodes 100A (Figure 2) and 100B are ordered. The 30 particular order of the quorum controllers is not particularly significant so long as both nodes 100A and 100B agree with respect to the particular ordering of the quorum controllers. In this illustrative example, shared storage device 112A is the first quorum controller, and shared storage device 112B is the second quorum controller. The ordering of the quorum controllers is used in a manner described more completely below to effect a second race for quorum if the first race for quorum results in a tie.

40 It is helpful in understanding the following description of logic flow diagram 300 (Figure 3) to remember that, in the event of a failure of communications link 210, both nodes 100A and 100B perform the steps of logic flow diagram 300 (Figure 3) independently as each detects the failure. Logic flow diagram 300 (Figure 3) is described in the context of node 100A. Processing according to logic flow diagram 300 (Figure 3) by node 100B (Figure 2) is directly analogous to the processing according to logic flow diagram 300 (Figure 3) by node 50 100A (Figure 2) as described below.

55 Steps 302-308 (Figure 3) represent a race for quorum between nodes 100A (Figure 2) and 100B. With step 302 (Figure 3), processing according to logic flow diagram 300 begins. In step 302, distributed database supervisor 202A (Figure 2) reserves on behalf of node 100A the quorum controller of the node in which distributed database supervisor 202A executes, i.e., node 100A. Reservation of a shared storage device by dis-

tributed database supervisor 202A is conventional and know but is described briefly for completeness.

Distributed database supervisor 202A reserves any one of shared storage devices 112A-D, e.g., shared storage device 112C, by issuing through processors 102A (Figure 1) a computer instruction, execution of which sends to shared storage device 112C signals requesting reservation of shared storage device 112C by node 100A. If shared storage device 112C is not reserved, shared storage device 112C changes its physical state to a reserved state in which storage device 112C sends to distributed database supervisor 202A through processors 102A signals indicating that shared storage device 112C is reserved by node 100A. Conversely, if shared storage device 112C is reserved when reservation requesting signals are received from processors 102A, shared storage device 112C sends to distributed database supervisor 202A through processors 102A signals indicating that the requested reservation of shared storage device 112C is refused. In this embodiment, reservations are held by a node, and distributed database supervisor 202A therefore requests reservations on behalf of node 100A.

In the context of distributed database supervisor 202A, the quorum controller of node 100A is called the "local quorum controller" and, in this illustrative example, is shared storage device 112A. Frequently, step 302 (Figure 3) will be a null operation, i.e., a step in which processors 102A (Figure 1) take no action, since node 100A (Figure 2) normally has the local quorum controller reserved. From step 302 (Figure 3), processing transfers to test step 304 in which distributed database supervisor 202A (Figure 2) determines whether the reservation of the local quorum controller is successful. As discussed more completely below, reservation of the local quorum controller can fail if distributed database supervisor 202B performs step 308 (Figure 3), which is described below, before distributed database supervisor 202A (Figure 2) performs step 302 (Figure 3). Under such circumstances, distributed database supervisor 202B (Figure 2) wins the race for quorum and can access shared storage devices 112A-D.

If distributed database supervisor 202A fails to reserve the local quorum controller, processing transfers from test step 304 (Figure 3) to step 306 in which distributed database supervisor 202A (Figure 2) aborts processing. Step 306 (Figure 3) is shown in greater detail as logic flow diagram 306 (Figure 4) in which processing begins in step 402. In step 402, distributed database supervisor 202A (Figure 2) relinquishes all reservations held by node 100A of any of shared storage devices 112A-D. Processing transfers to step 404 (Figure 4) in which distributed database supervisor 202A (Figure 4) denies all requests from distributed application 206A (Figure 2) for access to any of shared storage devices 112A-D. After step 404 (Figure 4), processing according to logic flow diagram 306, and therefore step 306 (Figure 3), terminates. After step 306, processing

according to logic flow diagram 300 terminates and distributed database supervisor 202A (Figure 2) has lost the race for quorum.

If, in test step 304 (Figure 3), distributed database

5 supervisor 202A (Figure 2) determines that the local quorum controller, i.e., shared storage device 112A in this illustrative example, is successfully reserved by node 100A, processing transfers to step 308 (Figure 3). In step 308, distributed database supervisor 202A (Figure 2) forcibly reserves on behalf of node 100A the quorum controller of the node in which distributed database supervisor 202A does not execute, i.e., node 100B. In the context of distributed database supervisor 202A, the quorum controller of node 100B is called the "remote quorum controller" and, in this illustrative example is shared storage device 112B. Ordinarily, attempted reservation of a device, which is already reserved by another computer process, fails. However, a forcible reservation of a device is a reservation which succeeds notwithstanding a prior reservation of the device which continues to be held by another node. Forcible reservations are known and are part of the Small Computer Serial Interface 3 (SCSI-3) standard established by the Common Access Method (CAM) Committee.

25 Thus, if node 100B itself or distributed database supervisor 202B has failed, i.e., if node 100B or distributed database supervisor 202B is now inoperative, distributed database supervisor 202A reserves for node 100A both quorum controllers in steps 302 (Figure 3) and 308.

30 This is true even if distributed database supervisor 202B (Figure 2) failed to relinquish the reservation of node 100B of its local quorum controller, e.g., shared storage device 112B, since distributed database supervisor 202A forcibly reserves on behalf of node 100A the local quorum controller of node 100B. Accordingly, failure to relinquish a reservation of a shared storage device when a node of distributed computer system 100 (Figure 1) fails cannot result in failure of distributed computer system 100 and the remaining operational node can acquire a quorum. It is appreciated that the local quorum controller of distributed database supervisor 202B (Figure 2) is the remote quorum controller of distributed database supervisor 202A and the remote quorum controller of distributed database supervisor 202B is the local quorum controller of distributed database supervisor 202A.

35 40 45 If node 100B and distributed database supervisor 202B continue to operate, steps 302-308 (Figure 3) represent a race to reserve both quorum controllers. As described above with respect to test step 304, it is possible that distributed database supervisor 202A (Figure 2) fails to reserve on behalf of node 100A its local quorum controller. Such a reservation fails if distributed database supervisor 202B (Figure 2) forcibly reserves its remote quorum controller, i.e., the local quorum controller of distributed database supervisor 202A, in performing step 308 (Figure 3) prior to performance of step 302 by distributed database supervisor 202A (Figure 2). In such

circumstances, distributed database supervisor 202B wins the race for quorum and distributed database supervisor 202A aborts in step 306 (Figure 3) as described more completely above.

If distributed database supervisor 202A (Figure 2) has won the race for quorum of steps 302 (Figure 3) through 308, node 100A (Figure 2) has reserved both the local quorum controller and the remote quorum controller, e.g., both shared storage devices 112A and 112B. Conversely, if distributed database supervisor 202A has lost the race for quorum, node 100B has reserved both the local and the remote quorum controller and processing by distributed database supervisor 202A terminates after performance of step 306 (Figure 3).

If the race for quorum between distributed database supervisors 202A (Figure 2) and 202B results in a tie, i.e., if distributed database supervisors 202A and 202B perform steps 302-308 (Figure 3) -- and step 308 in particular -- substantially simultaneously, each of nodes 100A-B has the remote quorum controller reserved but does not have the local quorum controller reserved. In this illustrative example, such a tie results in shared storage device 112B (Figure 2) reserved by node 100A and shared storage device 112A reserved by node 100B. In steps 310-312 (Figure 3), distributed database supervisor 202A (Figure 2) tests for such a tie, as does distributed database supervisor 202B in a separate, independent performance of steps 310-312 (Figure 3).

In step 310, distributed database supervisor 202A (Figure 2) attempts to reserve on behalf of node 100A its local quorum controller. In test step 312 (Figure 3), distributed database supervisor 202A (Figure 2) determines whether the attempted reservation of step 310 (Figure 3) is successful. If the reservation of the local quorum controller is successful, distributed database supervisor 202A (Figure 2) has won the race for quorum and processing according to logic flow diagram 300 (Figure 3) terminates. If distributed database supervisor 202A (Figure 2) has won the race for quorum, distributed database supervisor 202A can grant requests by distributed application 206A access to any of shared storage devices 112A-D. If, in test step 312 (Figure 3), distributed database supervisor 202A (Figure 2) determines that the attempted reservation of step 310 (Figure 3) failed, distributed database supervisor 202A (Figure 2) determines that the race for quorum of steps 302-308 (Figure 3) results in a tie and processing transfers from test step 312 to step 314.

Steps 314-324 represent a tie-breaking mechanism in which distributed database supervisors 202A (Figure 2) and 202B participate in a second race for quorum. Since step 314 (Figure 3) is performed only when a tie results in the first race for quorum, distributed database supervisors 202A (Figure 2) and 202B assume that both distributed database supervisors 202A and 202B are operative and the detected failure is in communications link 210. Accordingly, the second race for quorum can

use the reservation mechanism of shared storage devices 112A-D as an arbitration mechanism of the second race for quorum. In step 314 (Figure 3), distributed database supervisor 202A (Figure 2) causes node 100A to relinquish the reservation to the remote quorum controller such that node 100A no longer has either quorum controller reserved. In an analogous, independent performance of step 314 (Figure 3) by distributed database supervisor 202B (Figure 2), distributed database supervisor 202B causes node 100B to relinquish its remote quorum controller such that node 100B no longer has either quorum controller reserved. Processing transfers from step 314 (Figure 3) to step 316.

In step 316, distributed database supervisor 202A

(Figure 2) waits a randomly selected amount of time. Specifically, distributed database supervisor 202A generates a random number representing an amount of time within a predetermined range of time and distributed database supervisor 202A remains in an inoperative state during that randomly selected amount of time. Since the amount of time to wait is dependent upon a randomly generated number, distributed database supervisors 202A and 202B are likely to wait for different amounts of time and are therefore likely to finish respective independent performances of step 316 (Figure 3) at different times.

Processing transfers to step 318 in which distributed database supervisor 202A (Figure 2) attempts to re-reserve on behalf of node 100A the first quorum controller.

As described above, the quorum controllers are ordered and have the same ordering within both distributed database supervisors 202A and 202B. Thus, in analogous and separate performances of step 318 (Figure 3) by distributed database supervisors 202A (Figure 2) and 202B, both distributed database supervisor 202A and distributed database supervisor 202B attempt to re-reserve the same quorum controller, i.e., the same shared storage device. In this example, shared storage device 112A is the first quorum controller.

From step 318 (Figure 3), processing transfers to test step 320 in which distributed database supervisor 202A (Figure 2) determines whether the attempted reservation of step 318 (Figure 2) succeeded. If the attempted reservation failed, distributed database supervisor 202A (Figure 2) has lost the second race for quorum and processing transfers to step 306 (Figure 3) in which distributed database supervisor 202A (Figure 2) aborts as described more completely above. Conversely, if distributed database supervisor 202A determines

in test step 320 (Figure 3) that the attempted reservation of step 318 succeeded, processing transfers to step 322. At this point, distributed database supervisor 202A (Figure 2) can not be certain that distributed database supervisor 202A has acquired quorum. Since distributed database supervisors 202A and 202B execute independently and perhaps at different speeds, it is possible that distributed database supervisor 202A performs step 308 (Figure 3) just prior to performance of step 322,

which is described below, by distributed database supervisor 202B. In such a situation, node 100B has the second quorum controller, e.g., shared storage device 112B, reserved and distributed database supervisor 202A cannot therefore reserve both quorum controllers on behalf of node 100A and cannot therefore acquire quorum.

In step 322, distributed database supervisor 202A (Figure 2) attempts to reserve the second quorum controller, which is shared storage device 112B in this illustrative example. From step 322 (Figure 3), processing transfers to test step 324 in which distributed database supervisor 202A (Figure 2) determines whether the attempted reservation of step 320 (Figure 3) succeeded. If the attempted reservation failed, distributed database supervisor 202A (Figure 2) has lost the second race for quorum and processing transfers to step 306 (Figure 3) in which distributed database supervisor 202A (Figure 2) aborts as described more completely above. In such a situation, distributed database supervisor 202B proceeds with performance of steps 316-324 (Figure 3) and successfully acquires a quorum. Conversely, if distributed database supervisor 202A (Figure 2) determines in test step 324 (Figure 3) that the attempted reservation of step 322 succeeded, distributed database supervisor 202A (Figure 2) has won the second race for quorum and processing according to logic flow diagram 300 (Figure 3) terminates. As described above, if distributed database supervisor 202A (Figure 2) wins the race for quorum, distributed database supervisor 202A grants to distributed application 206A access to any of shared storage devices 112A-D.

Thus, in accordance with the steps of logic flow diagram 300 (Figure 3), distributed database supervisors 202A (Figure 2) and 202B independently achieve consensus regarding which of distributed database supervisors 202A and 202B acquires a quorum, and therefore access to shared storage devices 112A-D, without requiring that a failing node relinquish reservations during such failure. Accordingly, the present invention represents a substantial improvement in fault tolerance of two-node distributed computer systems over prior art mechanism.

#### Failure Detection

As described above, distributed database supervisor 202A (Figure 2) detects a failure of either node 100B or communications link 210. Detection of a failure of distributed computer system 100 by distributed database supervisor 202A is described briefly for completeness. Distributed database supervisor 202A detects the failure in a generally conventional manner, e.g., by observing that no message has been received from distributed database supervisor 202B in a predetermined amount of time. For example, distributed database supervisor 202A can be configured to send to distributed database supervisor 202B a status request signal which directs

distributed database supervisor 202B to respond with a signal containing data representing the status of distributed database supervisor 202B. If distributed database supervisor 202A does not receive such data from distributed database supervisor 202B within a predetermined amount of time, e.g., one second, from the time the status request signal is sent, a failure is assumed by distributed database supervisor 202A. Alternatively, distributed database supervisor 202B can be configured to send to distributed database supervisor 202A signal representing the status of distributed database supervisor 202B at predetermined intervals, e.g., every second. Failure to receive any such signals for a predetermined period of time, e.g., two or three times the predetermined interval, can be interpreted by distributed database supervisor 202A as a failure of communications link 210 or node 100B.

#### Claims

1. A method for granting to one of at least two nodes of a distributed computer system exclusive access to shared resources of the distributed computer system, which includes at least two preselected shared devices and in which each node includes a fault handler and has a state which specifies whether the node has exclusive access to the shared resources, the method comprising:

detecting a fault condition in the distributed computer system;  
causing each of the fault handlers to perform the following steps when a fault condition is detected:

attempting to reserve all of the preselected shared devices, at least one of which is not reserved by the node of the fault handler in the absence of a fault condition;  
causing the state of the node to specify that the node has exclusive access to the shared resources if the fault handler succeeds in reserving all of the preselected shared devices; and

causing the state of the node to specify that the node does not have exclusive access to the shared resources if the fault handler fails to reserve all of the preselected shared devices.

2. The method of Claim 1 wherein each of the fault handlers further perform the following step when a fault condition is detected:

detecting a tie condition if the node succeeds in reserving at least one of the preselected shared devices and fails in reserving at least one other of the preselected shared devices.

3. The method of Claim 2 wherein the distributed computer system includes a shared device which is predetermined to be a tie-breaking device;

further wherein each of the fault handlers further perform the following steps when a tie condition is detected:

attempting to reserve the tie-breaking device;

causing the state of the node to specify that the node has exclusive access to the shared resources if the fault handler succeeds in reserving the tie-breaking device; and

causing the state of the node to specify that the node does not have exclusive access to the shared resources if the fault handler fails in reserving the tie-breaking device.

4. The method of Claim 3 wherein the tie-breaking device is one of the preselected shared devices.

5. The method of Claim 3 wherein the distributed computer system includes a shared device which is predetermined to be a second tie-breaking device, which is different from the first-mentioned tie-breaking device;

further wherein each of the fault handlers further perform the following steps when the first tie-breaking device is successfully reserved by the fault handler:

attempting to reserve the second tie-breaking device;

causing the state of the node to specify that the node has exclusive access to the shared resources if the fault handler succeeds in reserving the second tie-breaking device; and

causing the state of the node to specify that the node does not have exclusive access to the shared resources if the fault handler fails in reserving the second tie-breaking device.

6. The method of Claim 5 wherein the second tie-breaking device is one of the preselected shared devices.

7. The method of Claim 1 further comprising:

designating for each node of the two-node distributed computer system a local shared device of the preselected shared devices;

wherein the step of attempting comprises:

5

attempting to reserve the local shared device of the node of the fault handler; and forcibly reserving at least one preselected shared device other than the local shared device of the node of the fault handler if the fault handler successfully reserves the local shared device of the node of the fault handler.

10 8. A computer program product comprising:

a computer usable medium having computer readable code embodied therein for granting to one of at least two nodes of a distributed computer system exclusive access to shared resources of the distributed computer system, which includes at least two preselected shared devices and in which each node includes a fault handler and has a state which specifies whether the node has exclusive access to the shared resources, the computer usable medium comprising:

a fault detection module which is configured to detect a fault condition in the distributed computer system; a fault handling module which is operatively coupled to the fault detection module and which is configured to perform the following steps when a fault condition is detected:

attempting to reserve all of the preselected shared devices, at least one of which is not reserved by the node of the fault handling module in the absence of a fault condition;

causing the state of the node to specify that the node has exclusive access to the shared resources if the fault handling module succeeds in reserving all of the preselected shared devices; and

causing the state of the node to specify that the node does not have exclusive access to the shared resources if the fault handling module fails to reserve all of the preselected shared devices.

25 9. The computer program product of Claim 8 wherein the fault handling module is further configured to perform the following step when a fault condition is detected:

detecting a tie condition if the node succeeds in reserving at least one of the preselected shared devices and fails in reserving at least one of the preselected shared devices.

50 55 10. The computer program product of Claim 9 wherein the fault handling module is further configured to

perform the following steps when a tie condition is detected:

attempting to reserve a shared tie-breaking device of the distributed computer system; causing the state of the node to specify that the node has exclusive access to the shared resources if the fault handling module succeeds in reserving the shared tie-breaking device; and causing the state of the node to specify that the node does not have exclusive access to the shared resources if the fault handling module fails in reserving the shared tie-breaking device.

11. The computer program product of Claim 10 wherein the shared tie-breaking device is one of the preselected shared devices.

12. The computer program product of Claim 10 wherein the fault handling module is further configured to perform the following steps when the shared tie-breaking device is successfully reserved by the fault handling module:

attempting to reserve a second shared tie-breaking device, which is different from the first-mentioned shared tie-breaking device; causing the state of the node of the fault handler to specify that the node of the fault handler has exclusive access to the shared resources if the fault handling module succeeds in reserving the second tie-breaking device; and causing the state of the node of the fault handler to specify that the node of the fault handler does not have exclusive access to the shared resources if the fault handling module fails in reserving the second tie-breaking device.

13. The computer program product of Claim 12 wherein the second tie-breaking device is one of the preselected shared devices.

14. The computer program product of Claim 8 wherein the fault handling module is configured to perform the step of attempting by performing at least the following steps:

attempting to reserve a local shared device of the preselected shared devices; and forcibly reserving at least one preselected shared devices other than the local shared device if the fault handling module successfully reserves the local shared device.

15. An apparatus for recovering from a failure in a distributed computer system which includes shared re-

sources and at least two nodes, the apparatus comprising:

5 a fault detector which corresponds to a selected one of the nodes of the distributed computer system and which detects a fault condition in the distributed computer system; a fault handler (i) which is operatively coupled to the fault detector, (ii) which has a state which specifies whether the selected node has exclusive access to the shared resources, and (iii) which, upon detection of the fault condition:

10 (a) attempts to reserve all of the preselected shared devices, at least one of which is not reserved by the selected node in the absence of a fault condition; (b) causes the state of the fault handler to specify that the selected node has exclusive access to the shared resources if the fault handler succeeds in reserving all of the preselected shared devices; and (c) causes the state of the fault handler to specify that the selected node does not have exclusive access to the shared resources if the fault handler fails to reserve all of the preselected shared devices.

15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110 115 120 125 130 135 140 145 150 155 160 165 170 175 180 185 190 195 200 205 210 215 220 225 230 235 240 245 250 255 260 265 270 275 280 285 290 295 300 305 310 315 320 325 330 335 340 345 350 355 360 365 370 375 380 385 390 395 400 405 410 415 420 425 430 435 440 445 450 455 460 465 470 475 480 485 490 495 500 505 510 515 520 525 530 535 540 545 550 555 560 565 570 575 580 585 590 595 600 605 610 615 620 625 630 635 640 645 650 655 660 665 670 675 680 685 690 695 700 705 710 715 720 725 730 735 740 745 750 755 760 765 770 775 780 785 790 795 800 805 810 815 820 825 830 835 840 845 850 855 860 865 870 875 880 885 890 895 900 905 910 915 920 925 930 935 940 945 950 955 960 965 970 975 980 985 990 995 1000 1005 1010 1015 1020 1025 1030 1035 1040 1045 1050 1055 1060 1065 1070 1075 1080 1085 1090 1095 1100 1105 1110 1115 1120 1125 1130 1135 1140 1145 1150 1155 1160 1165 1170 1175 1180 1185 1190 1195 1200 1205 1210 1215 1220 1225 1230 1235 1240 1245 1250 1255 1260 1265 1270 1275 1280 1285 1290 1295 1300 1305 1310 1315 1320 1325 1330 1335 1340 1345 1350 1355 1360 1365 1370 1375 1380 1385 1390 1395 1400 1405 1410 1415 1420 1425 1430 1435 1440 1445 1450 1455 1460 1465 1470 1475 1480 1485 1490 1495 1500 1505 1510 1515 1520 1525 1530 1535 1540 1545 1550 1555 1560 1565 1570 1575 1580 1585 1590 1595 1600 1605 1610 1615 1620 1625 1630 1635 1640 1645 1650 1655 1660 1665 1670 1675 1680 1685 1690 1695 1700 1705 1710 1715 1720 1725 1730 1735 1740 1745 1750 1755 1760 1765 1770 1775 1780 1785 1790 1795 1800 1805 1810 1815 1820 1825 1830 1835 1840 1845 1850 1855 1860 1865 1870 1875 1880 1885 1890 1895 1900 1905 1910 1915 1920 1925 1930 1935 1940 1945 1950 1955 1960 1965 1970 1975 1980 1985 1990 1995 2000 2005 2010 2015 2020 2025 2030 2035 2040 2045 2050 2055 2060 2065 2070 2075 2080 2085 2090 2095 2100 2105 2110 2115 2120 2125 2130 2135 2140 2145 2150 2155 2160 2165 2170 2175 2180 2185 2190 2195 2200 2205 2210 2215 2220 2225 2230 2235 2240 2245 2250 2255 2260 2265 2270 2275 2280 2285 2290 2295 2300 2305 2310 2315 2320 2325 2330 2335 2340 2345 2350 2355 2360 2365 2370 2375 2380 2385 2390 2395 2400 2405 2410 2415 2420 2425 2430 2435 2440 2445 2450 2455 2460 2465 2470 2475 2480 2485 2490 2495 2500 2505 2510 2515 2520 2525 2530 2535 2540 2545 2550 2555 2560 2565 2570 2575 2580 2585 2590 2595 2600 2605 2610 2615 2620 2625 2630 2635 2640 2645 2650 2655 2660 2665 2670 2675 2680 2685 2690 2695 2700 2705 2710 2715 2720 2725 2730 2735 2740 2745 2750 2755 2760 2765 2770 2775 2780 2785 2790 2795 2800 2805 2810 2815 2820 2825 2830 2835 2840 2845 2850 2855 2860 2865 2870 2875 2880 2885 2890 2895 2900 2905 2910 2915 2920 2925 2930 2935 2940 2945 2950 2955 2960 2965 2970 2975 2980 2985 2990 2995 3000 3005 3010 3015 3020 3025 3030 3035 3040 3045 3050 3055 3060 3065 3070 3075 3080 3085 3090 3095 3100 3105 3110 3115 3120 3125 3130 3135 3140 3145 3150 3155 3160 3165 3170 3175 3180 3185 3190 3195 3200 3205 3210 3215 3220 3225 3230 3235 3240 3245 3250 3255 3260 3265 3270 3275 3280 3285 3290 3295 3300 3305 3310 3315 3320 3325 3330 3335 3340 3345 3350 3355 3360 3365 3370 3375 3380 3385 3390 3395 3400 3405 3410 3415 3420 3425 3430 3435 3440 3445 3450 3455 3460 3465 3470 3475 3480 3485 3490 3495 3500 3505 3510 3515 3520 3525 3530 3535 3540 3545 3550 3555 3560 3565 3570 3575 3580 3585 3590 3595 3600 3605 3610 3615 3620 3625 3630 3635 3640 3645 3650 3655 3660 3665 3670 3675 3680 3685 3690 3695 3700 3705 3710 3715 3720 3725 3730 3735 3740 3745 3750 3755 3760 3765 3770 3775 3780 3785 3790 3795 3800 3805 3810 3815 3820 3825 3830 3835 3840 3845 3850 3855 3860 3865 3870 3875 3880 3885 3890 3895 3900 3905 3910 3915 3920 3925 3930 3935 3940 3945 3950 3955 3960 3965 3970 3975 3980 3985 3990 3995 4000 4005 4010 4015 4020 4025 4030 4035 4040 4045 4050 4055 4060 4065 4070 4075 4080 4085 4090 4095 4100 4105 4110 4115 4120 4125 4130 4135 4140 4145 4150 4155 4160 4165 4170 4175 4180 4185 4190 4195 4200 4205 4210 4215 4220 4225 4230 4235 4240 4245 4250 4255 4260 4265 4270 4275 4280 4285 4290 4295 4300 4305 4310 4315 4320 4325 4330 4335 4340 4345 4350 4355 4360 4365 4370 4375 4380 4385 4390 4395 4400 4405 4410 4415 4420 4425 4430 4435 4440 4445 4450 4455 4460 4465 4470 4475 4480 4485 4490 4495 4500 4505 4510 4515 4520 4525 4530 4535 4540 4545 4550 4555 4560 4565 4570 4575 4580 4585 4590 4595 4600 4605 4610 4615 4620 4625 4630 4635 4640 4645 4650 4655 4660 4665 4670 4675 4680 4685 4690 4695 4700 4705 4710 4715 4720 4725 4730 4735 4740 4745 4750 4755 4760 4765 4770 4775 4780 4785 4790 4795 4800 4805 4810 4815 4820 4825 4830 4835 4840 4845 4850 4855 4860 4865 4870 4875 4880 4885 4890 4895 4900 4905 4910 4915 4920 4925 4930 4935 4940 4945 4950 4955 4960 4965 4970 4975 4980 4985 4990 4995 5000 5005 5010 5015 5020 5025 5030 5035 5040 5045 5050 5055 5060 5065 5070 5075 5080 5085 5090 5095 5100 5105 5110 5115 5120 5125 5130 5135 5140 5145 5150 5155 5160 5165 5170 5175 5180 5185 5190 5195 5200 5205 5210 5215 5220 5225 5230 5235 5240 5245 5250 5255 5260 5265 5270 5275 5280 5285 5290 5295 5300 5305 5310 5315 5320 5325 5330 5335 5340 5345 5350 5355 5360 5365 5370 5375 5380 5385 5390 5395 5400 5405 5410 5415 5420 5425 5430 5435 5440 5445 5450 5455 5460 5465 5470 5475 5480 5485 5490 5495 5500 5505 5510 5515 5520 5525 5530 5535 5540 5545 5550 5555 5560 5565 5570 5575 5580 5585 5590 5595 5600 5605 5610 5615 5620 5625 5630 5635 5640 5645 5650 5655 5660 5665 5670 5675 5680 5685 5690 5695 5700 5705 5710 5715 5720 5725 5730 5735 5740 5745 5750 5755 5760 5765 5770 5775 5780 5785 5790 5795 5800 5805 5810 5815 5820 5825 5830 5835 5840 5845 5850 5855 5860 5865 5870 5875 5880 5885 5890 5895 5900 5905 5910 5915 5920 5925 5930 5935 5940 5945 5950 5955 5960 5965 5970 5975 5980 5985 5990 5995 6000 6005 6010 6015 6020 6025 6030 6035 6040 6045 6050 6055 6060 6065 6070 6075 6080 6085 6090 6095 6100 6105 6110 6115 6120 6125 6130 6135 6140 6145 6150 6155 6160 6165 6170 6175 6180 6185 6190 6195 6200 6205 6210 6215 6220 6225 6230 6235 6240 6245 6250 6255 6260 6265 6270 6275 6280 6285 6290 6295 6300 6305 6310 6315 6320 6325 6330 6335 6340 6345 6350 6355 6360 6365 6370 6375 6380 6385 6390 6395 6400 6405 6410 6415 6420 6425 6430 6435 6440 6445 6450 6455 6460 6465 6470 6475 6480 6485 6490 6495 6500 6505 6510 6515 6520 6525 6530 6535 6540 6545 6550 6555 6560 6565 6570 6575 6580 6585 6590 6595 6600 6605 6610 6615 6620 6625 6630 6635 6640 6645 6650 6655 6660 6665 6670 6675 6680 6685 6690 6695 6700 6705 6710 6715 6720 6725 6730 6735 6740 6745 6750 6755 6760 6765 6770 6775 6780 6785 6790 6795 6800 6805 6810 6815 6820 6825 6830 6835 6840 6845 6850 6855 6860 6865 6870 6875 6880 6885 6890 6895 6900 6905 6910 6915 6920 6925 6930 6935 6940 6945 6950 6955 6960 6965 6970 6975 6980 6985 6990 6995 7000 7005 7010 7015 7020 7025 7030 7035 7040 7045 7050 7055 7060 7065 7070 7075 7080 7085 7090 7095 7100 7105 7110 7115 7120 7125 7130 7135 7140 7145 7150 7155 7160 7165 7170 7175 7180 7185 7190 7195 7200 7205 7210 7215 7220 7225 7230 7235 7240 7245 7250 7255 7260 7265 7270 7275 7280 7285 7290 7295 7300 7305 7310 7315 7320 7325 7330 7335 7340 7345 7350 7355 7360 7365 7370 7375 7380 7385 7390 7395 7400 7405 7410 7415 7420 7425 7430 7435 7440 7445 7450 7455 7460 7465 7470 7475 7480 7485 7490 7495 7500 7505 7510 7515 7520 7525 7530 7535 7540 7545 7550 7555 7560 7565 7570 7575 7580 7585 7590 7595 7600 7605 7610 7615 7620 7625 7630 7635 7640 7645 7650 7655 7660 7665 7670 7675 7680 7685 7690 7695 7700 7705 7710 7715 7720 7725 7730 7735 7740 7745 7750 7755 7760 7765 7770 7775 7780 7785 7790 7795 7800 7805 7810 7815 7820 7825 7830 7835 7840 7845 7850 7855 7860 7865 7870 7875 7880 7885 7890 7895 7900 7905 7910 7915 7920 7925 7930 7935 7940 7945 7950 7955 7960 7965 7970 7975 7980 7985 7990 7995 8000 8005 8010 8015 8020 8025 8030 8035 8040 8045 8050 8055 8060 8065 8070 8075 8080 8085 8090 8095 8100 8105 8110 8115 8120 8125 8130 8135 8140 8145 8150 8155 8160 8165 8170 8175 8180 8185 8190 8195 8200 8205 8210 8215 8220 8225 8230 8235 8240 8245 8250 8255 8260 8265 8270 8275 8280 8285 8290 8295 8300 8305 8310 8315 8320 8325 8330 8335 8340 8345 8350 8355 8360 8365 8370 8375 8380 8385 8390 8395 8400 8405 8410 8415 8420 8425 8430 8435 8440 8445 8450 8455 8460 8465 8470 8475 8480 8485 8490 8495 8500 8505 8510 8515 8520 8525 8530 8535 8540 8545 8550 8555 8560 8565 8570 8575 8580 8585 8590 8595 8600 8605 8610 8615 8620 8625 8630 8635 8640 8645 8650 8655 8660 8665 8670 8675 8680 8685 8690 8695 8700 8705 8710 8715 8720 8725 8730 8735 8740 8745 8750 8755 8760 8765 8770 8775 8780 8785 8790 8795 8800 8805 8810 8815 8820 8825 8830 8835 8840 8845 8850 8855 8860 8865 8870 8875 8880 8885 8890 8895 8900 8905 8910 8915 8920 8925 8930 8935 8940 8945 8950 8955 8960 8965 8970 8975 8980 8985 8990 8995 9000 9005 9010 9015 9020 9025 9030 9035 9040 9045 9050 9055 9060 9065 9070 9075 9080 9085 9090 9095 9100 9105 9110 9115 9120 9125 9130 9135 9140 9145 9150 9155 9160 9165 9170 9175 9180 9185 9190 9195 9200 9205 9210 9215 9220 9225 9230 9235 9240 9245 9250 9255 9260 9265 9270 9275 9280 9285 9290 9295 9300 9305 9310 9315 9320 9325 9330 9335 9340 9345 9350 9355 9360 9365 9370 9375 9380 9385 9390 9395 9400 9405 9410 9415 9420 9425 9430 9435 9440 9445 9450 9455 9460 9465 9470 9475 9480 9485 9490 9495 9500 9505 9510 9515 9520 9525 9530 9535 9540 9545 9550 9555 9560 9565 9570 9575 9580 9585 9590 9595 9600 9605 9610 9615 9620 9625 9630 9635 9640 9645 9650 9655 9660 9665 9670 9675 9680 9685 9690 9695 9700 9705 9710 9715 9720 9725 9730 9735 9740 9745 9750 9755 9760 9765 9770 9775 9780 9785 9790 9795 9800 9805 9810 9815 9820 9825 9830 9835 9840 9845 9850 9855 9860 9865 9870 9875 9880 9885 9890 9895 9900 9905 9910 9915 9920 9925 9930 9935 9940 9945 9950 9955 9960 9965 9970 9975 9980 9985 9990 9995 10000 10005 10010 10015 10020 10025 10030 10035 10040 10045 10050 10055 10060 10065 10070 10075 10080 10085 10090 10095 10100 10105 10110 10115 10120 10125 10130 10135 10140 10145 10150 10155 10160 10165 10170 10175 10180 10185 10190 10195 10200 10205 10210 10215 10220 10225 10230 10235 10240 10245 10250 10255 10260 10265 10270 10275 10280 10285 10290 10295 10300 10305 10310 10315 10320 10325 10330 10335 10340 10345 10350 10355 10360 10365 10370 10375 10380 10385 10390 10395 10400 10405 10

of the fault handler to specify that the selected node has exclusive access to the shared resources if the fault handler succeeds in reserving the second tie-breaking device; and (b) causes the state of the fault handler to specify that the selected node does not have exclusive access to the shared resources if the fault handler fails in reserving the second tie-breaking device.

20. The apparatus of Claim 19 wherein the second tie-breaking device is one of the preselected shared devices.

21. The apparatus of Claim 15 wherein the fault handler attempts to reserve all of the preselected shared devices by (i) attempting to reserve a local shared device of the preselected shared devices and (ii) forcibly reserving at least one the local shared device if the fault handler successfully reserves the local shared device.

22. A distributed computer system comprising:

shared resources which include at least two preselected shared devices; at least two nodes, each of which is operatively coupled to each of the preselected shared devices and each of which comprises:

at least one processor; a network access device which is operatively coupled to the at least one processor and through which the at least one processor can communicate with at least one processor of another of the at least two nodes; a fault detector which detects a fault condition in which the node can no longer communicate with at least one other of the at least two nodes; a fault handler (i) which is operatively coupled to the fault detector, (ii) which has a state which in turn specifies whether the node has exclusive access to the shared resources, and (iii) which, upon detection of the fault condition:

(a) attempts to reserve all of the preselected shared devices, at least one of which is not reserved by the fault handler in the absence of a fault condition; (b) causes the state of the fault handler to specify that the node has exclusive access to the shared resources if the fault handler succeeds in reserving all of the preselected shared devices; and (c) causes the state of the fault handler to specify that the node does not have exclusive access to the shared resources if the fault handler fails to reserve all of the preselected shared devices.

5 23. The distributed computer system of Claim 22 wherein, in the fault condition, the fault handler of each node detects a tie condition if the fault handler succeeds in reserving at least one of the preselected shared devices and fails in reserving at least one other of the preselected shared devices.

10 24. The distributed computer system of Claim 23 wherein, when the fault handler of each node detects the tie condition, the fault handler (i) attempts to reserve a shared tie-breaking device of the distributed computer system and (ii)(a) causes the state of the fault handler to specify that the node has exclusive access to the shared resources if the fault handler succeeds in reserving the shared tie-breaking device and (b) causes the state of the fault handler to specify that the node has exclusive access to the shared resources if the fault handler fails in reserving the shared tie-breaking device.

15 25. The distributed computer system of Claim 24 wherein the shared tie-breaking device is one of the preselected shared devices.

20 26. The distributed computer system of Claim 24 wherein, when the shared tie-breaking device is successfully reserved by the fault handler of one of the nodes, the fault handler of the node (i) attempts to reserve a second shared tie-breaking device, which is different from the first-mentioned shared tie-breaking device; (ii)(a) causes the state of the fault handler to specify that the node has exclusive access to the shared resources if the fault handler of the node succeeds in reserving the second tie-breaking device; and (b) causes the state of the fault handler to specify that the node does not have exclusive access to the shared resources if the fault handler of the node fails in reserving the second tie-breaking device.

25 27. The distributed computer system of Claim 26 wherein the second tie-breaking device is one of the preselected shared devices.

30 28. The distributed computer system of Claim 22 wherein the fault handler of each node attempts to reserve all of the preselected shared devices by (i) attempting to reserve a selected one of the preselected shared devices which is local to the node and (ii) forcibly reserving at least one of the preselected shared devices other than the selected shared device if the fault handler successfully reserves the selected shared device.

35

40

45

50

55

29. A system for distributing code (i) which is stored on a computer-readable medium, (ii) which is executable by a computer, and (iii) which includes at least one module, each of which in turn is configured to carry out at least one function to be executed by the computer, the system comprising:

a fault detection module which is configured to detect a fault condition in a distributed computer system which includes at least two nodes and shared resources which in turn include at least two preselected shared devices;  
a fault handling module (i) which corresponds to a selected one of the at least two nodes, (ii) which is operatively coupled to the fault detection module, and (iii) which is configured to perform the following steps when a fault condition is detected:

attempting to reserve for the selected node all of the preselected shared devices, at least one of which is not reserved by the node of the fault handling module in the absence of a fault condition;  
causing a state of the selected node to specify that the selected node has exclusive access to the shared resources if the fault handling module succeeds in reserving all of the preselected shared devices; and  
causing the state of the selected node to specify that the selected node does not have exclusive access to the shared resources if the fault handler fails to reserve all of the preselected shared devices.

30. The system of Claim 29 wherein the fault handling module is further configured to perform the following step when a fault condition is detected:

detecting a tie condition if the fault handling module succeeds in reserving for the selected node at least one of the preselected shared devices and fails in reserving for the selected node at least one others of the preselected shared devices.

31. The system of Claim 30 wherein the fault handling module is further configured to perform the following steps when a tie condition is detected:

attempting to reserve for the selected node a shared tie-breaking device of the distributed computer system;  
causing the state of the selected node to specify that the selected node has exclusive access to the shared resources if the fault handling module succeeds in reserving the shared tie-breaking device; and  
causing the state of the selected node to spec-

ify that the selected node does not have exclusive access to the shared resources if the fault handling module fails in reserving the shared tie-breaking device.

5 32. The system of Claim 31 wherein the shared tie-breaking device is one of the preselected shared devices.

10 33. The system of Claim 31 wherein the fault handling module is further configured to perform the following steps when the shared tie-breaking device is successfully reserved by the fault handling module:

15 attempting to reserve for the selected node a second shared tie-breaking device, which is different from the first-mentioned shared tie-breaking device;

causing the state of the selected node of the fault handler to specify that the selected node of the fault handler has exclusive access to the shared resources if the fault handling module succeeds in reserving the second tie-breaking device; and

causing the state of the selected node of the fault handler to specify that the selected node of the fault handler does not have exclusive access to the shared resources if the fault handling module fails in reserving the second tie-breaking device.

30 34. The system of Claim 33 wherein the second tie-breaking device is one of the preselected shared devices.

35 35. The system of Claim 29 wherein the fault handling module is configured to perform the step of attempting by performing at least the following steps:

40 attempting to reserve for the selected node a local shared device of the preselected shared devices; and

45 forcibly reserving for the selected node at least one preselected shared devices other than the local shared device if the fault handling module successful reserves the local shared device.

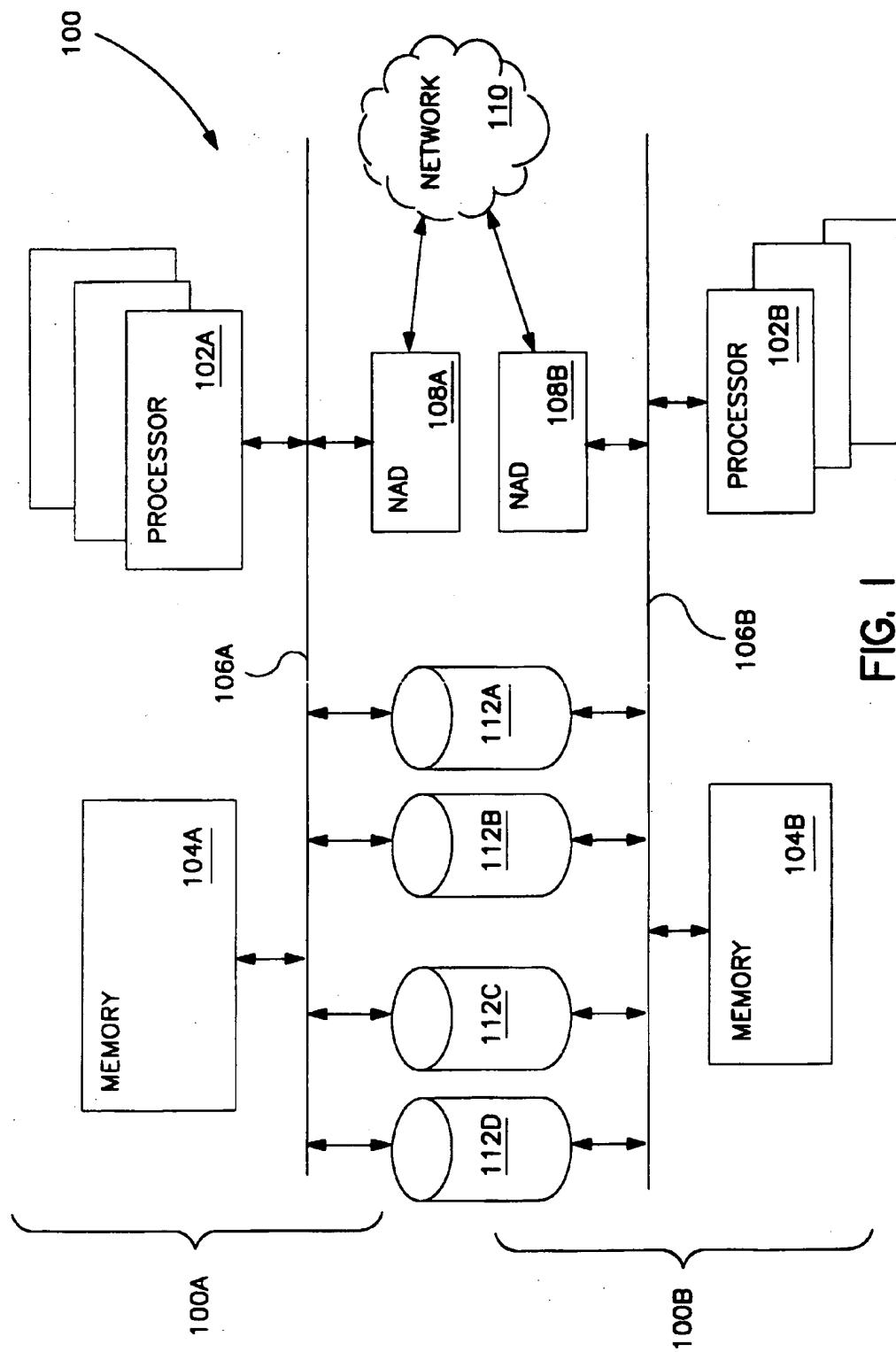
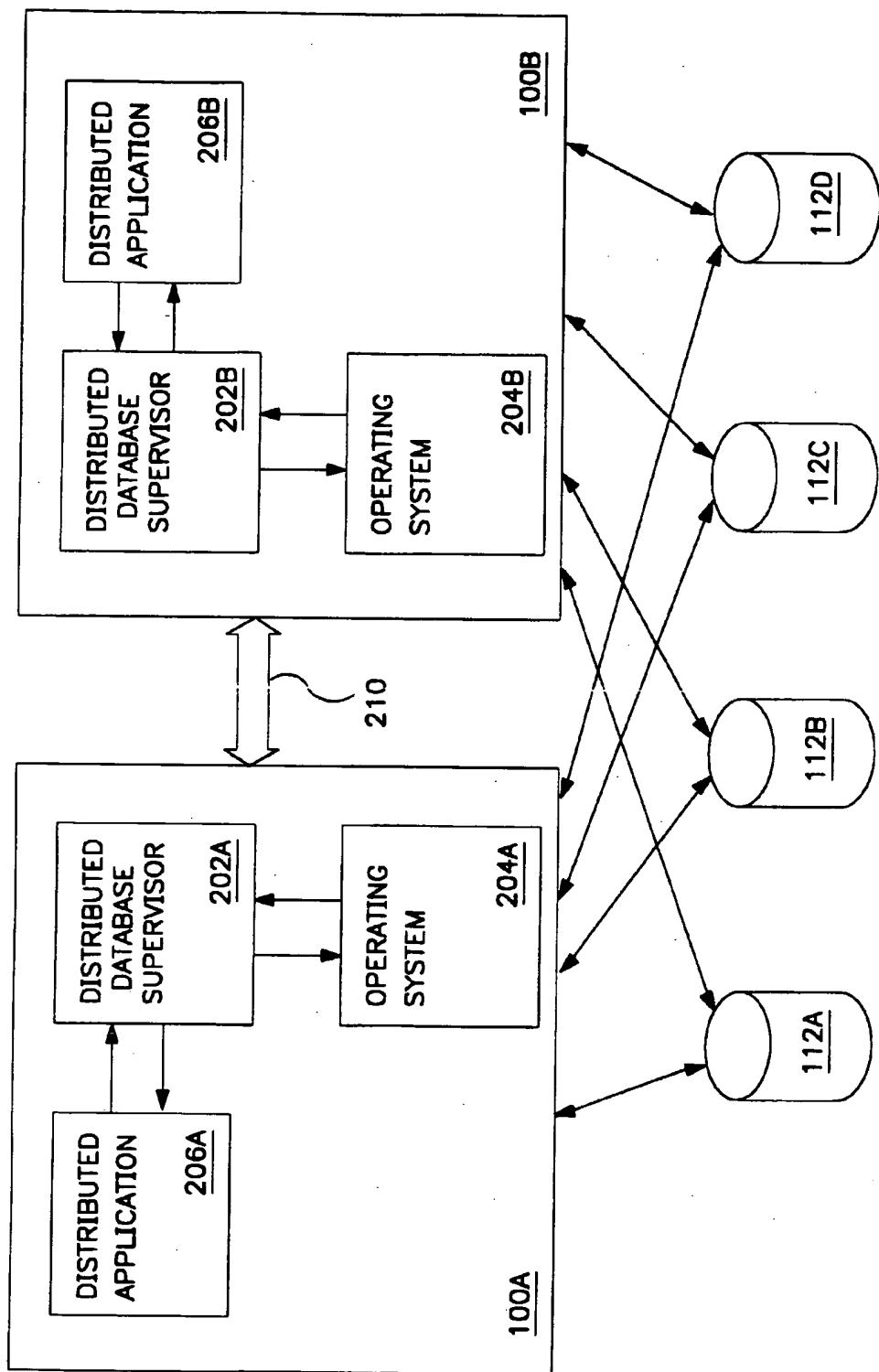


FIG. I



2  
FIG.

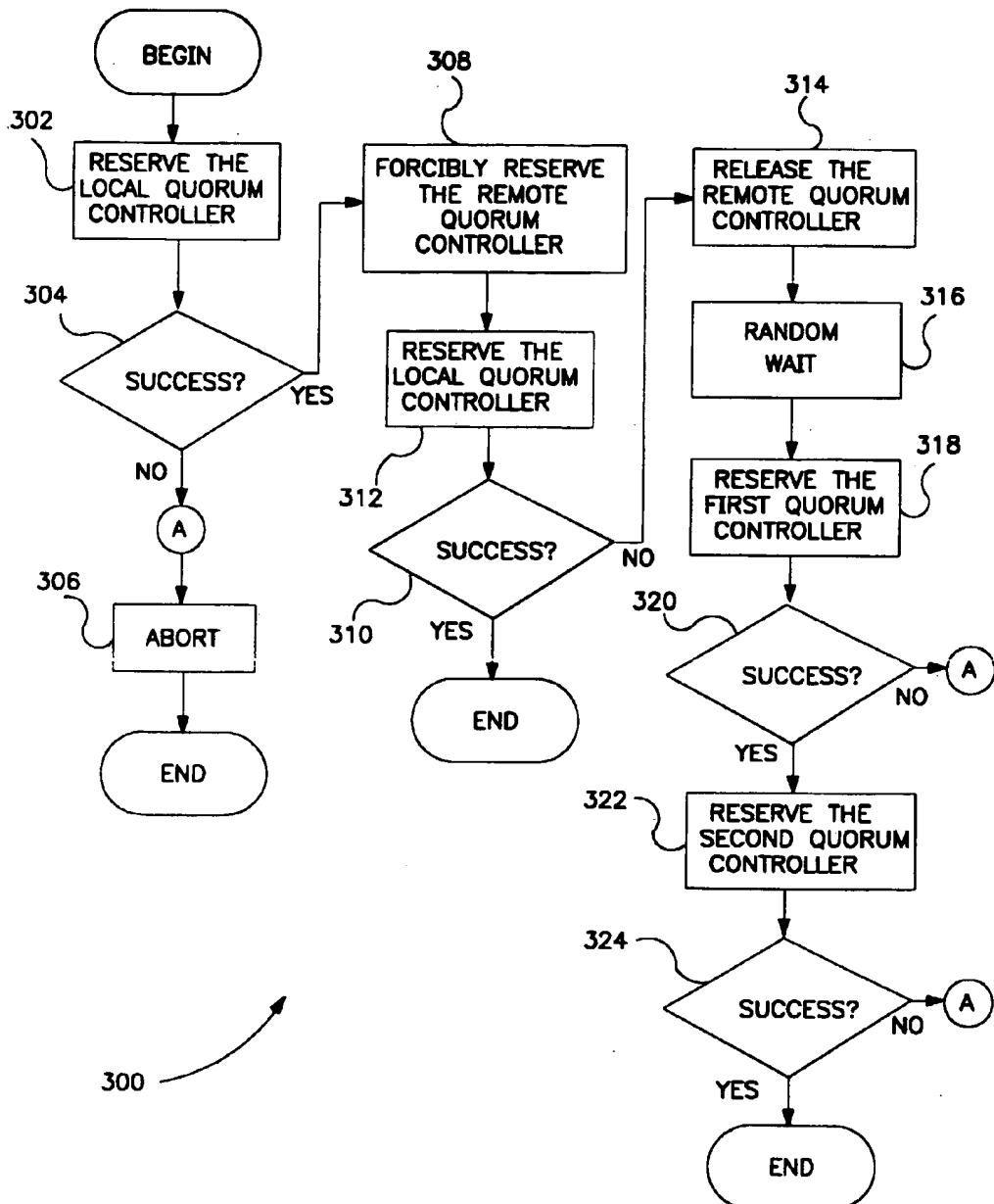


FIG. 3

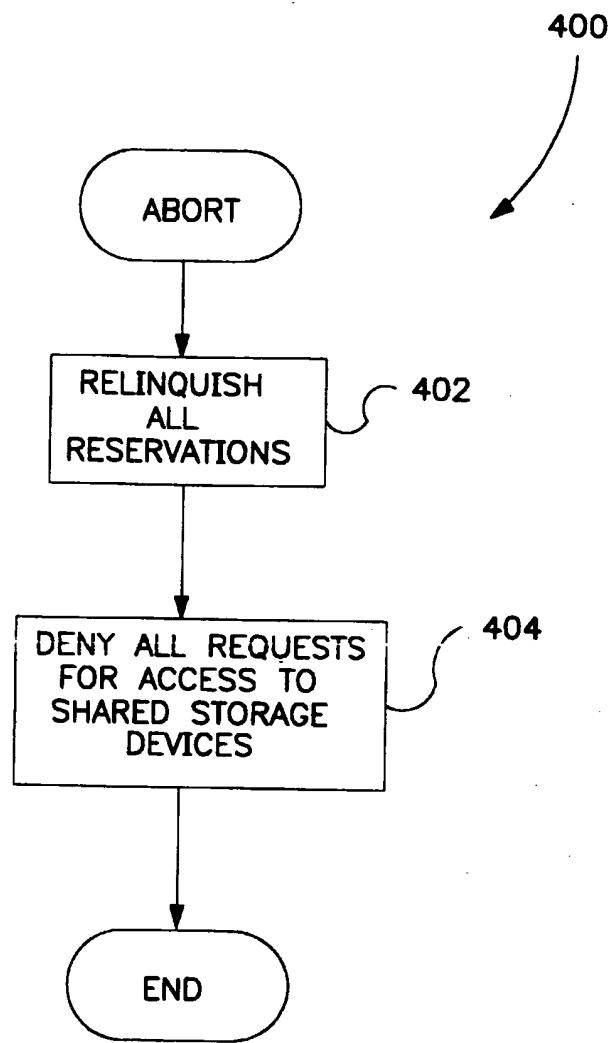


FIG. 4



European Patent  
Office

## EUROPEAN SEARCH REPORT

Application Number  
EP 97 30 3680

DOCUMENTS CONSIDERED TO BE RELEVANT												
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.)									
A	EP 0 595 453 A (INTERNATIONAL COMPUTERS LIMITED) * page 13, line 46 - page 15, line 43 *	1-35	G06F11/14									
A	EP 0 537 899 A (SUN MICROSYSTEMS, INC.) * abstract *	1-35										
A	EP 0 033 915 A (HONEYWELL INFORMATION SYSTEMS ITALIA S.P.A.) * page 4, line 13 - line 23 *	1-35										
-----												
TECHNICAL FIELDS SEARCHED (Int.Cl.)												
G06F												
<p>The present search report has been drawn up for all claims</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;">Place of search</td> <td style="width: 33%;">Date of completion of the search</td> <td style="width: 33%;">Examiner</td> </tr> <tr> <td>THE HAGUE</td> <td>9 September 1997</td> <td>Corremans, G</td> </tr> <tr> <td colspan="3"> <b>CATEGORY OF CITED DOCUMENTS</b> <div style="display: flex; justify-content: space-between; font-size: small;"> <div style="width: 45%;"> X : particularly relevant if taken alone  Y : particularly relevant if combined with another document of the same category  A : technological background  O : non-written disclosure  P : intermediate document </div> <div style="width: 45%;"> T : theory or principle underlying the invention  E : earlier patent document, but published on, or after the filing date  D : document cited in the application  L : document cited for other reasons  &amp; : member of the same patent family, corresponding document </div> </div> </td> </tr> </table>				Place of search	Date of completion of the search	Examiner	THE HAGUE	9 September 1997	Corremans, G	<b>CATEGORY OF CITED DOCUMENTS</b> <div style="display: flex; justify-content: space-between; font-size: small;"> <div style="width: 45%;"> X : particularly relevant if taken alone  Y : particularly relevant if combined with another document of the same category  A : technological background  O : non-written disclosure  P : intermediate document </div> <div style="width: 45%;"> T : theory or principle underlying the invention  E : earlier patent document, but published on, or after the filing date  D : document cited in the application  L : document cited for other reasons  &amp; : member of the same patent family, corresponding document </div> </div>		
Place of search	Date of completion of the search	Examiner										
THE HAGUE	9 September 1997	Corremans, G										
<b>CATEGORY OF CITED DOCUMENTS</b> <div style="display: flex; justify-content: space-between; font-size: small;"> <div style="width: 45%;"> X : particularly relevant if taken alone  Y : particularly relevant if combined with another document of the same category  A : technological background  O : non-written disclosure  P : intermediate document </div> <div style="width: 45%;"> T : theory or principle underlying the invention  E : earlier patent document, but published on, or after the filing date  D : document cited in the application  L : document cited for other reasons  &amp; : member of the same patent family, corresponding document </div> </div>												

**THIS PAGE BLANK (USPTO)**